Results Document:

What I Achieved:

During this development period I was able to build a Database Driven Web Application Quiz Manager. With the functionality of:

* Create quizzes which will be stored within the database.
* Edit existing quizzes and store save changes in the database.
* The Ability for the user to Join completed quizzes where they will need to answer or multiple choice answers and be given a score at the end of the test showing how many they got right
* The ability to personalise the appearance of the application, such as being able to change the colours of the application.
* A login system where the passwords are hashed and stored within the database.
* Two different levels of permissions being “Student” and “Teacher”.

When first loading up the Quiz Manager the user will be brought to the Login Screen, The user will be asked to enter their first name, surname and Password. Once the users enters their information into the textboxes and press the “login” button the information they have entered are compared to data stored in the database in order to verify the credentials are correct as well as what permissions the user should have.

The passwords that are used are hashed using SHA-256 and when the user enters their password, the password is then also hashed and compared to the existing hashed password stored on the database to check if it’s identical.

Graphical user interface

Description automatically generated

For Creating the Quiz. This functionality is only available through the use of a “Teacher” account. This is done by the user selecting the “Create” option on the options bar on the left hand side of the screen where they will be taken to a screen which will allow them to create a new or load and existing one, indicated by the “New” and “Load” Buttons.

Graphical user interface, text

Description automatically generated

Once the “New” button is pressed the user will be taken to a new screen where they will be asked to enter a name for the quiz as well as what subject the test is for.

Graphical user interface, website

Description automatically generated

Once the user has entered this information they will be taken the the actual quiz creation page.

Graphical user interface

Description automatically generated

The user is able to add more questions to the quiz my pressing the “Add” button and remove questions from the quiz by pressing the “Del” button, however there must be at least one question for each quiz. The user can then navigate between the questions my pressing the “Next” and “Previous”.

Above all these options will be the 6 Digit Code that is randomly assigned to the quiz, this value will need to be recorded somewhere as it is how the Teacher will be able to edit the quiz as well as how students will access the quiz.

In the middle of the screen the user will have textboxes where they can enter the description for the question as well as the four multiple choice answers for the question. There will also be a radio button next to the answers where the one selected will be the correct answer.

Joining a Quiz is available for both the “Student” and “Teacher” account permissions and is the only option available for the “Student” account. This option will be the default screen that the user will see once logging in, however on a “Teacher” account it can be navigated back to by pressing the “Join” option on the options bar on the left hand side of the screen.

Graphical user interface, website

Description automatically generated

The screen will ask the user for a 6 Digit Code which will be given out to the students externally. Once the join button the user will be taken to the Quiz Answering Page.

Chart, treemap chart

Description automatically generated

When loaded onto the Answering Page the layout is very similar to the creation page in terms of it’s layout. The user is able to navigate between the questions using the “Next” and “previous” buttons and are able to complete the questions in any order they want, with the number of questions they have completed and remaining located above the navigation buttons.

How the users will answer the question is they will be given the Description of the question as well as four different answers with radio buttons next to each one, the user will then select the radio button they think is the correct answer and press the “Save” button. Once all of the questions are answered a “Finish” button will appear which will take the user to the results screen.

Chart, treemap chart

Description automatically generated

The results screen will then show the user how many they got right out of how many questions there are in total. There will then be a button which will take the user back to the “Join” page.

The Settings Page is only available to “Teacher” accounts and is accessible by pressing the “Join” option on the options bar on the left hand side of the screen.

Chart, treemap chart

Description automatically generated

On this screen the user will be able to change the colours of the website be selecting the corresponding box which will bring up a colour selector where the user can select the colour they want.

Once all the colour changes the user wants is selected the user can then hit the “Apply” button which will apply the changes to the Web App.

Limitations of my Design:

What I felt were some of the limitations of my design was some of the dependencies that I used were Windows only, which would mean that porting the Web App to operating systems such as Linux or MacOS would require several parts of the software to be redesigned in order to remove the dependencies on those Windows libraries.

Another area that I felt shows some of the limitations of my design is the Web App was only designed for Computers and Laptops in mind and not Mobile Devices. Although the App was designed to support scaling for multiple types of aspect ratios and resolutions, it was designed to support the layout of Mobile Devices as well as a touchscreen input.

Additionally, some other limitations of my design is that the only form of input for creating quizzes is text and doesn’t support the ability to upload photos or videos. Which is due to the database structure I used.

Proposed Future Improvements:

If I had more time to work on this project one of the changes I would make is allowing a more easier method to edit quizzes. As in my current Quiz Manager the only way to edit an existing quiz is to remember the “Join Code” and if you join code is forgotten or lost then you lose access the the Quiz. What I planed on doing was having a scrollable table on the “Create” Page which allowed you to select and edit existing quizzes removing the need for the Teachers to remember the code. This was however not implemented due to lack of time and my inexperience with auto generating tables in HTML.

Another area for improvement would be the security. As my Quiz Manager uses SQL statements in order to access and update data from the database. However the Quiz App has basically no protection from SQL Injections. If I were to continue developing this software further I would focus on adding protection from SQL Injection attacks.

Additionally, If I had more time to work on this project I would finish some of the unfinished areas I planned to add but was cut because they weren’t required for the project, an example was a “Manage Class” page when the Teachers could view all the Students and there class information. There was also a feature on the on the “Create” page where the user could edit the most recently accessed quiz, this was partially worked on but wasn’t finished due to more important areas needing completing.